METHOD FOR INTERFACING SCANNED PRODUCT INFORMATION WITH A SOURCE FOR THE PRODUCT OVER A GLOBAL NETWORK

TECHNICAL FIELD OF THE INVENTION

This invention relates in general to a method for controlling a computer and, more particularly, to a method for modifying scanned bar code data associated with a manufacturer's product for input to a computer over a global communication network to a manufacturer's location to effect a link therebetween.

CROSS REFERENCE TO RELATED APPLICATION

This application is related to copending U.S. Patent Application Serial No. 09/151,530, entitled "METHOD FOR CONTROLLING A COMPUTER WITH AN AUDIO SIGNAL," (Atty Dkt No. PHLY-24,398) filed of even date herewith.

BACKGROUND OF THE INVENTION

With the advent of the global information network, commonly referred to as the "Internet," there has been an increase in the number of marketing locations that have been disposed on the network to take advantage of the interconnectivity provided therewith to a user. These marketing locations allow users to access various products by accessing the "website" of the particular marketing location which in effect creates a "virtual mall." In order to provide this access, the user is required to open up a "browser" application on their computer, which allows them access to locations on the web. Each location on the network is defined in terms of a Universal Resource Locator ("URL") such that the user need only input this URL into a web browser to effect a connection with that associated location. The user typically inputs a "domain" name which is sent to the network for conversion to the URL of the owner of that domain name. The use of the domain name has increased the ease of use on the network, as each location on the web can now have an easily decipherable address.

The type of information that is provided to a user from a given marketing location can be an interactive transaction which allows the user to select products, determine pricing, and even order products. Some of these marketing locations provide access to many products. However, other locations provide access to only a single company and all of its products. The primary problem with marketing products over the Internet or providing information over the Internet is to somehow inform the user of the location and connect that user to that particular location via an associated URL or domain name. At present, the user is required to type the domain or URL into a browser window and then instruct the browser to find that location over the network. This presents several disadvantages. First, the user somehow needs to have information about a particular company or one of its products such that there is a desire on the part of the user to seek more information. Second, there must be some type of link between the product and the "web page" of a particular company in order to allow the user to easily go from the product to the web page to retrieve more information about the product. At present, there is no easily accessible system that can link a product with the web page of the manufacturer of that product.

SUMMARY OF THE INVENTION

The present invention disclosed and claimed herein comprises a method for associating a product with remote information for that product. A product identifier is accessed at a source location on a network which is in close association with a product such that it identifies the associated product. Routing information is appended to the accessed product identifier, which routing information defines the location of a remote location on the network . The product information is then transmitted from the source location to the remote location over the network in accordance with the appended routing information, which routing information defines the network connection between the source location and the remote location on the network, wherein information regarding the associated product can be returned to the source location from the remote location.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following description taken in conjunction with the accompanying Drawings in which:

FIGURE 1 illustrates a block diagram of a computer system using the proprietary wedge interface;

FIGURE 2 illustrates the general I/O of the microcontroller used in the proprietary wedge electronics;

FIGURE 3 illustrates the retrieval system for obtaining product information over a global communication network;

FIGURES 4a-4e illustrate the various message packets transmitted between the source PC and servers used in the preferred embodiment;

FIGURE 5 illustrates a flowchart of the method for receiving and encoding the data prior to transmission to an Advertiser Reference Server ("ARS");

FIGURE 6 illustrates a flowchart of actions taken by the ARS server;

FIGURE 7 illustrates a flowchart of the interactive process between the source computer and ARS;

FIGURE 8 illustrates a web browser page receiving the modified URL/bar code data according to the preferred embodiment;

FIGURE 9 illustrates a simplified block diagram of the disclosed embodiment;

FIGURE 10 illustrates a more detailed, simplified block diagram of the embodiment of FIGURE 9;

FIGURE 11 illustrates a diagrammatic view of a method for performing the routing operation;

FIGURE 12 illustrates a block diagram of an alternate embodiment utilizing an audio input for generating the routing information;

FIGURE 13 illustrates a block diagram illustrating the generation of a profile with the disclosed embodiment;

FIGURE 14 illustrates a flowchart for generating the profile and storing it at the ARS;

FIGURE 15 illustrates a flowchart for processing the profile information when information is routed to a user.

DETAILED DESCRIPTION OF THE INVENTION

Referring now to FIGURE 1, there is illustrated a Personal Computer ("PC") 108 connected to receive inputs from a bar code scanner wand 100 and a keyboard 106, and also connected through a network interface 112 to a network. The PC 108 runs a browser program to facilitate the access of information on the network, for example, a global communication network known as the "Internet" or the World-Wide-Web ("Web"). The browser is a hypertext-linked application used for accessing information. Hypertext is a term used to describe a particular organization of information within a data processing system, and its presentation to a user. It exploits the computer's ability to link together information from a wide variety of sources to provide the user with the ability to explore a particular topic. The traditional style of presentation used in books employs an organization of the information which is imposed upon it by limitations of the medium, namely fixed sized, sequential paper pages. Hypertext systems, however, use a large number of units of text or other types of data such as image information, graphical information, video information, or sound information, which can vary in size. A collection of such units of information is termed a hypertext document, or where the hypertext documents employ information other than text, hypermedia documents.

Multimedia communications may use the Hypertext Transfer Protocol ("HTTP"), and files or formatted data may use the Hypertext Markup Language ("HTML"). This formatting language provides for a mingling of text, graphics, sound, video, and hypertext links by "tagging" a text document using HTML. Data encoded using HTML is often referred to as an "HTML document," an "HTML page," or a "home page." These documents and other Internet resources may be accessed across the network by means of a network addressing scheme which uses a locator referred to as a Uniform Resource Locator ("URL"), for example, "http://www.digital.com."

The Internet is one of the most utilized networks for interconnecting distributed computer systems and allows users of these computer systems to exchange data all over the world. Connected to the Internet are many private networks, for example, corporate or commercial networks. Standard protocols, such as the Transport Control Protocol ("TCP") and the Internet Protocol ("IP") provide a convenient method for communicating across these

diverse networks. These protocols dictate how data are formatted and communicated. As a characteristic of the Internet, the protocols are layered in an IP stack. At higher levels of the IP stack, such as the application layer (where HTTP is employed), the user information is more readily visible, while at lower levels, such as the network level (where TCP/IP are used), the data can merely be observed as packets or a stream of rapidly moving digital signals. Superimposed on the Internet is a standard protocol interface for accessing Web resources, such as servers, files, Web pages, mail messages, and the like. One way that Web resources can be accessed is by browsers made by *Netscape*® and *Microsoft Internet Explorer*®.

Referring again now to FIGURE 1, during a scanning mode, scanner 100 scans bar code data 102 into a "wedge" interface 104 for conversion into keyboard data, which keyboard data is passed therefrom into the keyboard input port of PC 108. A wedge interface 104 is simply an interface box containing circuitry that accommodates inputs from both a scanner and a computer keyboard 106. Therefore output from the scanner 100 is passed into the wedge 104 for conversion into keyboard data. Since the wedge 104 emulates a keyboard, the PC 108 does not require a separate input for the scanner 100, but may pass scanner data through the wedge 104 directly into the keyboard port of the PC 108. When not receiving scanner data, the wedge 104 simply acts as a pass-through device for keyboard data. The information is ultimately presented to a viewer by display 110.

In operation, a code for a specific product is provided in the form of a bar code 102. This bar code is the "link" to a product. The disclosed embodiment is operable to provide a link between that product and the web page of the manufacturer of that product by utilizing the bar code as routing information over the net, while also providing for an accounting step, as will be described in more detail hereinbelow. The bar code 102 by itself is not compatible with any kind of network for the purposes of communication therewith. It is primarily provided for a retail-type setting. Therefore, the information contained in the bar code 102, by itself, does not allow for anything other than identification of a product, assuming that one has a database containing information as to a correlation between the product and the bar code.

The information in the bar code 102 is first fed into the wedge interface 104, which wedge interface 104 is operable to decode the bar code 102, and then "append" to that bar code intermediate routing information. This intermediate routing information is information that determines where the bar code information is to be sent on the network. This will be routed to the network through a network interface 112.

When assembling the intermediate routing information in the form of the bar code specifically, and the information defining the location of the ARS 308 on the network, it is important to note that the information in the bar code must be converted from the optical image to the numerical values input to the keyboard and then converted into HTML language. The reason for this is that it must be converted into a browser compatible format for interface with the browser. Of course, if another type of program were utilized, then it would have to converted into language compatible with that program.

The wedge interface 104, once having decoded the bar code information 102 and appending the intermediate routing information thereto, then organizes the bar code/intermediate routing information into keystroke data which is then routed to the PC 108. The reason for this is that the interface 104 merely makes the wand scanner 100, the information scanned, and the appended intermediate routing information appear as keystrokes to the PC 108, which essentially emulates a user on the keyboard 106 inputting the same information. Therefore, the combination of the interface 104 and the wand 100 allows a user to "key" data into the PC 108 through the use of the coded information in the bar code 102. Of course, a user could actually key in the information on the bar code 102 and then append the appropriate intermediate routing information associated therewith. As will be described hereinbelow, the intermediate routing information appended thereto is the URL of the destination server on the network.

Referring now to FIGURE 2, there is illustrated a microcontroller 200 which controls interfacing of the keyboard 106 and wand 100 with the PC 108. The microcontroller 200 has contained therein a memory 202 or it can have external memory. There are provided a plurality of interfaces 204 to the wand 100, a plurality of interfaces 206 to the PC 108, and plurality of

interfaces 208 to the keyboard 106. In general, the wand 100 requires a serial data line, a ground line, and a power line. Similarly, the keyboard 106 requires a serial data line, a ground line, a clock line, and a power line. The PC 108 provides a clock line, a power line, a serial data, and a ground line for input to the microcontroller 200. The microcontroller 200 is operable to receive signals from the keyboard 106 and transfer the signals to the PC 108 as keyboard signals. Operation with the keyboard 106 is essentially a "pass-through" procedure. Data output from the keyboard is already in keyboard format, and therefore requires no conversion by the wedge interface 104. With respect to the wand 100, the serial data is not compatible with a keyboard 106 and, therefore, it must be translated into a keyboard format in order to allow input thereof to the keyboard input of the PC 108. The microcontroller 200 performs this function after decoding this bar code information and conversion of this bar code information into an appropriate stream of data which is comprised of the bar code information and the appended URL. This appended URL will be pre-stored in the memory 202 and is programmable at the time of manufacture. It is noted that the memory 202 is illustrated as being contained within the microcontroller 202 to provide a single chip solution. However, this could be external memory that is accessible by the microcontroller 202. Therefore, the microcontroller 200 provides an interface between a wand 100 and a keyboard 106 to a PC 108 which allows the wand 100 to receive coded information and convert it to keyboard strokes with appended intermediate routing information or, alternatively, to merely pass-through the keystrokes from the keyboard 106. Therefore, the user need not install any type of plug-in circuit board into the motherboard of the computer in order to provide an interface to the wand 100; rather, the user need only utilize the already available keyboard port in order to input the appropriate data into the system.

In this particular disclosed embodiment, the microcontroller 200 comprises a PIC16C73 microcontroller by *Microchip Technologies*™. The PIC16C73 device is a low cost CMOS 8-bit microcontroller with an integrated analog-to-digital converter. The PIC16C73 device, as illustrated in the disclosed embodiment, has 192 bytes of RAM and 4k x 4 of EPROM memory. The microcontroller 200 can accommodate asynchronous or synchronous inputs from input devices connected to it. In this disclosed embodiment,

communication to the keyboard 106 is synchronous while it is asynchronous when communicating with wand 100.

It should be noted that, although in this particular embodiment bar code information 102 is input into the keyboard input of the PC 108, disclosed methods may also be advantageously utilized with high speed port architectures such as Universal Serial Bus ("USB") and IEEE 1394.

Bar codes are structured to be read in either direction. Timing considerations need to be addressed because of the variety of individuals scanning the bar code introduce a wide variety of scan rates. Bar codes use bars of varying widths. The presence of a black bar generates a positive pulse, and the absence of a black bar generates no pulse. Each character of a conventional bar code has associated therewith seven pulses or bars. Depending on the width of the bars, the time between pulses varies. In this disclosed embodiment, the interface circuitry 104 performs a "running" calculation of the scan time based upon the rising edge of the pulses commencing with the leader or header information. The minimum and maximum scans times are calculated continuously in software with the interface 104 during the scanning process to ensure a successful scan by the user.

Referring now to FIGURE 3, there is illustrated as a network source a PC 302, similar to PC 108, connected to a network interface 304, similar to network interface 112, which in turn connects to a global communication network 306. In this embodiment, data scanned by wand 100 is received by PC 302 with appended intermediate routing information as emulated keystrokes. This information directs PC 302 to obtain information from an Advertiser Reference Server ("ARS") 308 over Path "A." The ARS 308 is a network device disposed on the network that is defined as the location to which the scanned data from the bar code 102 is to be routed. This is defined by the routing information programmed into the interface 104. As such, scanned information will always be routed to the ARS 308, since a URL is unique on the system. Connected to the ARS 308 is a database 310 of product codes and associated manufacturers. The database 310 undergoes a continual update process which is transparent to the user. As companies sign-on to the system disclosed herein, i.e., subscribe thereto,

manufacturer and product information is added to the database 310 without interrupting operation of the source PC 302 with frequent updates.

Referring now to FIGURE 4a, the message packet 400 sent from source PC 302 to ARS 308 via Path "A" comprises several fields. One field comprises the URL of the ARS 308. Another field comprises the advertiser product code or other information derived from the bar code 102, and any additional overhead information required for a given transaction. Yet another field comprises the network address of the source PC 302. In general, network transmissions are effected in packets of information, each packet providing a destination address, a source address, and data. These packets vary depending upon the network transmission protocol utilized for communication. Although the protocols utilized in the disclosed embodiment are of a conventional protocol suite commonly known as TCP/IP, it should be understood that any protocols providing the similar basic functions can be used, with the primary requirement that a browser can forward the bar code information to the desired URL in response to keystrokes being input to a PC. Hereinbelow, "message packet" shall refer to the combination of the destination URL, bar code information, and source address, even though more than a single packet must be transmitted to effect such a transmission.

Upon receipt of the message packet 400 from source PC 302, ARS 308 processes the information in accordance with instructions embedded in the overhead information. The ARS 308 specifically will extract the bar code information from the received packet 400 and, once extracted, will then decode this bar code information. Once decoded, this information is then compared with data contained within the ARS advertiser database 310 to determine if there is a "hit." If there is no "hit" indicating a match, then information is returned to the browser indicating such. If there is a "hit," a packet 402 is assembled which comprises the address of the source PC 302, and information instructing the source PC 302 as to how to access, directly in a "handoff" operation, another location on the network, that of an advertiser server 312. This type of construction is relatively conventional with browsers such as *Netscape*® and *Microsoft Internet Explorer*® and, rather than displaying information from the ARS 308, the source PC 302 can then access the advertiser server 312. The ARS 308 transmits the packet 402 back to source PC 302 over Path "B." Referring now to FIGURE 4b, the message

packet 402 comprises the address of the source PC 302, the URL of the advertiser server 312 embedded within instructional code, and the URL of the ARS 308.

Upon receipt of the message packet 402 by the source PC 302, the message packet 402 is disassembled to obtain pertinent routing information for assembly of a new message packet 404. The web browser running on source PC 302 is now directed to obtain, over Path "C," the product information from the particular advertiser server 312 location information embedded in message packet 404. Referring now to FIGURE 4c, the message packet 404 for this transaction comprises the URL of the advertiser server 312, the request-for-product-information data, and the address of the source PC 302.

Upon receipt of the message packet 404 from source PC 302, advertiser server 312 disassembles the message packet 404 to obtain the request-for-product-information data. The advertiser server 312 then retrieves the particular product information from its database, and transmits it over Path "D" back to the source PC 302. Referring now to FIGURE 4d, the message packet 406 for this particular transaction comprises the address of the source PC 302, the requested information, and the URL of the advertiser server 312.

Optionally, the ARS 308 may make a direct request for product information over Path "E" to advertiser server 312. In this mode, the ARS 308 sends information to the advertiser server 312 instructing it to contact the source PC 302. This, however, is unconventional and requires more complex software control. The message packet 408 for this transaction is illustrated in FIGURE 4e, which comprises the URL of the advertiser server 312, the request-for-product-information data, and the address of the source PC 302. Since product information is not being returned to the ARS 308, but directly to the source PC 302, the message packet 408 requires the return address to be that of the source PC 302. The product information is then passed directly to PC 302 over Path "D."

Referring now to FIGURE 5, there is illustrated a flowchart depicting the process for assembling the message packet 400 transmitted from the source PC 302 to the ARS 308. The process begins in function block 500 wherein the UPC bar code 102 is scanned by the user. In

function block 502, the scanner wand 102 interprets the coded bars and transmits the signal over the serial connection to the wedge interface 104 where the bar code data is then decoded by software stored in firmware. The decoded data is then stored in a word register in function block 504, for easy manipulation. Program flow continues to function block 506 where the word register data is then converted into keystroke data and stored. At this point, the product information is stored in memory on the wedge interface 104. From this point, information is sent serially through the keyboard port to be inserted into the web browser. In function block 508, a Ctrl-O command is added to the keystroke data. The hypertext protocol preamble http:// is added to the keystroke data in a function block 510. From there the program flows to function block 512 where the URL of the ARS 308 is added to the keystroke data. In function block 514, characters /? are added to the keystroke data. In function block 516, the UPC code data is added to the keystroke data. In function block 518, a carriage return is added to the keystroke data to effect transmission of the data to the ARS 308 in function block 520, over Path "A."

It should be noted that this embodiment is not limited to UPC bar code standards, but may also incorporate other bar code standards. For example, if a Code 128 standard were used, optional steps may be employed to deal with a system which uses this particular code, or multiple bar code standards. Function block 522 may be used in a multi-code scenario to determine the particular bar code standard used. Function block 524 enables the software to handle the particular standard being employed.

Referring now to FIGURE 6, there is illustrated a flowchart of the process the ARS 308 may undergo when receiving the message packet 400 from the source PC 302. In decision block 600, the ARS 308 checks for the receipt of the message packet 400. If a message packet 400 is not received, program flow moves along the "N" path to continue waiting for the message. If the message packet 400 is received, program flow continues along path "Y" for message processing. Upon receipt of the message packet 400, in function block 602, the ARS 308 extracts the characters /? and the following variable field comprising the UPC code. The UPC code is then extracted independently in function block 604 in preparation for matching the product code with the appropriate advertiser server address. In function block 606, the UPC

code is then used with a look-up table to retrieve the advertiser server 312 URL of the respective product information contained in the UPC code. In function block 608, the ARS 308 then assembles message packet 402 for transmission back to the source PC 302. Function block 610 indicates the process of sending the message packet 402 back to the source PC 302 over Path "B."

Referring now to FIGURE 7, there is illustrated a flowchart of the interactive processes between the source PC 302 and the advertiser server 312. In function block 700, the source PC 302 receives the message packet 402 back from the ARS 308 and begins to decode the packet 402. In function block 702, the URL of the advertiser product information is extracted from the message packet 402 and saved for insertion into the message packet 404 to the advertiser server 312. The message packet 404 is then assembled and sent by the source PC 302 over Path "C" to the advertiser server 312, in function block 704. While the source PC 302 waits, in function block 706, the advertiser server 312 receives the message packet 404 from the source PC 302, in function block 708, and disassembles it. The product information location is then extracted from the message packet 404 in function block 710. The particular product information is retrieved from the server database for transmission back to the source PC 302. In function block 712, the product information is assembled into message packet 406 and then transmitted back to the source PC 302 over Path "D." Returning to the source PC 302 in function block 714, the advertiser product information contained in the message packet 406 received from the advertiser server 312, is then extracted and processed in function block 716.

Referring now to FIGURE 8, a user launches a web browser application on a source PC 302 and computer display 800 presents a browser page 802. The user then scans the bar code 102 with a scanner wand 100. The product information embedded in the keystroke data is transmitted into the wedge interface 104 where proprietary URL data is appended to the bar code data to create an appended word. The appended word, in the format of keystroke data, is transmitted from the wedge interface 104 into the keyboard input port of source PC 302. Upon receipt of the URL/bar code keystroke data 808 from the wedge interface 104, a URL page 804 is opened. URL page 804 is opened in response to a Ctrl-O command added by the

proprietary wedge interface 104 as the first character string. Opening URL page 804 automatically positions the cursor in a field 806 where additional keystroke data following the Ctrl-O command will be inserted. After URL page 804 is opened, the hypertext protocol preamble http:// is inserted into the field 806. Next, URL information associated with the location of the ARS 308 is inserted into field 806. Following the ARS 308 URL data are the characters /? to allow entry of variables immediately following the /? characters. In this embodiment, the variable following is a UPC product code which provides the cross-reference information for obtaining the advertiser URL from the ARS database 310. Next, a carriage return is added to send the URL/bar code data and close the window 804. After the message packet 400 is transmitted to the ARS 308 from the source PC 302, transactions from the ARS 308, to the source PC 302, to the advertiser server 312, and back to the source PC 302, occur quickly and are transparent to the viewer. At this point, the next information the viewer sees is the product information which was received from the advertiser server 312.

Referring now to FIGURE 9, there is illustrated a product 902 which can be either a product, an advertisement ("ad"), or some tangible or visible device to which the user has access. In association with this product or ad 902, there is provided in this disclosed embodiment a region 904 which has contained therein the primary routing information. In the embodiment described hereinabove, this primary routing information is in the form of a bar code. However, it should be understood that, from an optical standpoint, the primary routing information could be encoded in any type of medium. For example, there could be provided an optically encoded background which, when scanned, would allow information to be extracted therefrom which would constitute primary routing information. One example of this is the digital encoding via optical means of a background on a driver's license. By scanning the entire driver's license, digital information embedded within the optical coding is extracted to provide primary routing information. Many other examples are present wherein optical information is encoded onto some tangible medium.

Once the primary routing information is extracted from the region 904, it is then input to the PC which, as disclosed hereinabove, utilizes the scanning wand 100 and the interface 104. The primary routing information, after being extracted from the region 904, through either the

decoding associated with the wand 100 or some other technique, is then input to a PC 906, similar to the PC 108 described hereinabove. The PC 906 is interfaced with a user 908, which user 908 is operable to initiate the transfer of the routing information from the region 904 to the PC 906. This is an active system which requires some type of user intervention. As described hereinabove, the PC 906, in response to receiving the primary routing information from the region 904, will then access a network 910 to provide an interconnection medium with a plurality of network nodes (not shown), one of these having associated therewith remote information on a network node 912. This network node 912 has contained therein remote information that is associated with the primary routing information 904. The PC 906, in conjunction with the network 910, is operable to access the remote information 912 via the primary routing information extracted from the optical field 904 such that there is connection made between the PC 906 and the remote information node 912 to allow information to then be extracted from the remote information node 912 and provided to the user 908. This is all accomplished by the user actively inputting the information in the optical field 904 to the PC 906 via some type of input device. In the disclosed embodiment, this does require the user to actually launch a web browser and then allow the input device, i.e., the wand 100, to input the information to the web browser. However, software within the PC 906 could be implemented to actually launch the browser in response to input of information from the optical field 904.

Referring now to FIGURE 10, there is illustrated a more detailed diagrammatic view of the embodiment of FIGURE 9. The routing information in the region 904 is input to a PC 1002, the PC 1002 being similar to the PC 108 in FIGURE 1 and also to the PC 906. However, in the detailed embodiment, the PC 1002 is operable to assemble a packet of information comprised of secondary routing information and the primary routing information in field 904 for transfer thereto. The secondary routing information is routing information that the PC 1002 uses to transfer the primary routing information in field 904 to an intermediate PC 1004 over the network 910. This intermediate PC 1004 is basically ARS 308 in FIGURE 3. Associated with the PC 1004 is the database 1006, similar to the advertiser database 310. However, this database can contain other information. The operation, as described hereinabove, requires a user 908 to first access the routing information in field 904 by some user initiated operation. Once input to the PC 1002, the primary routing information in field 904 is assembled with

secondary routing information which then forwards this primary routing information in field 904 to the PC 1004. The PC 1004 looks up information in its database 1006 in the form of table to determine where the location of the remote information is on the network 910. This is then relayed back to the PC 1002 which then effects a connection to the remote information node 912. Therefore, it can be seen that with the use of PC 1004 and its associated database 1006, information in the optical field 904 can be related to network routing information to the remote node 912. Note that, in order for the PC 1002 to be interfaced with the remote information network node 912, it is necessary to transfer all of the information in the primary routing information field 904 in the form of the primary routing information to the PC 1004 in order effect a connection.

Referring now to FIGURE 11, there is illustrated a diagrammatic view of how the network packet is formed for sending the primary routing information to the PC 1004. In general, the primary routing information occupies a single field which primary routing information is then assembled into a data packet with the secondary routing information for transfer to the network 910. This is described hereinabove in detail. However, it is important to recognize that the information contained within the optical field 904 in the form of a bar code, an optically encoded region, etc., constitutes routing information. For example, once the database 1006 at the PC 1004 has been programmed with the manufacturer's location, the manufacturer knows that it only needs to place this routing information in an optical field in association with either a product or an advertisement or some other medium in order to allow the user access thereto. It is the association of this optical field with the product that encourages the user to input the optical information to a network to view additional information about the product. Of course, the additional information is then viewed by the user or accessible by the user as defined by the manufacturer and not by the PC 1004. The PC 1004 merely relays this information.

Referring now to FIGURE 12, there is illustrated a block diagram of an alternate embodiment of the present invention utilizing an audio input source. In the embodiment of FIGURE 12, there is provided a program media 1202, which program media is operable to contain therein some type of routing information in the form of an audio signal. This can be an encoded tone, or other such device which, when coupled with an audio output device 1204 will

cause routing information in the form of an audio tone to be input to the PC 906. The PC 906 has associated therewith a conventional sound card which is operable to receive the audio signal on a dedicated port therefor and digitize this tone for processing by the PC 906 internal thereto. This decoding operation is a conventional operation. However, the audio output device 1204 is a device that is activated by the user 908. This operates similar to the embodiment of FIGURE 9 in that the user must activate it and, therefore, this is an active operation. Once received, in response thereto, the routing information from remote information node 912 is provided to the user. The program media 1202 and the audio output device 1204 could be any type of media and audio output device. One type of media might be that associated with children's books which have the ability to access audio information about the picture.

Referring now to FIGURE 13, there is illustrated a block diagram for an embodiment wherein a user's profile can be forwarded to the original subscriber or manufacturer. The PC 906 has associated therewith a profile database 1302, which profile database 1302 is operable to store a profile of the user 908. This profile is created when the program, after initial installation, requests profile information to be input in order to activate the program. In addition to the profile, there is also a unique ID that is provided to the user in association with the browser program that runs on the PC 906. This is stored in a storage location represented by a block 1304. This ID 1304 is accessible by a remote location as a "cookie" which is information that is stored in the PC 906 in an accessible location, which accessible location is actually accessible by the remote program running on a remote node.

The ARS 308, which basically constitutes the PC 1004 of FIGURE 10, is operable to have associated therewith a profile database 1308, which profile database 1308 is operable to store profiles for all of the users. The profile database 1308 is a combination of the profiles stored in profile database 1302 for all of the PCs 906 that are attachable to the system. This is to be distinguished from information stored in the database 310, the advertiser's database, which contains intermediate destination tables. When the routing information in the field 904 is forwarded to the ARS 308 and extracted from the original data packet, the look-up procedure described hereinabove can then be performed to determine where this information is to be routed. The profile database is then utilized for each transaction, wherein each transaction in the

form of the routing information received from the field 904 is compared to the destination tables 310 to determine what manufacturer it is associated with, and the associated ID that is transmitted along with the routing information in field 904 is then compared with the profile database 1308 to determine if a profile associated therewith is available. This information is stored in a transaction database 1310 such that, at a later time, for each routing code or bar code received in the form of the information in field 904, there will associated therewith the IDs of each of the PCs 906. The associated profiles in database 1308, which are stored in association with IDs, can then be assembled and transmitted to a subscriber as referenced by a subscriber node 1312 on the network 910. The ARS 308 can do this in two modes, a realtime mode or a non-realtime mode. In a realtime mode, each time a PC 906 accesses the advertiser database 310, that user's profile information is uploaded to the subscriber node 1312. At the same time, billing information is generated for that subscriber 1312 which is stored in a billing database 1316. Therefore, the ARS 308 has the ability to inform the subscriber 1312 of each transaction, bill for those transactions, and also provide to the subscriber 1312 profile information regarding who is accessing the particular product advertisement having associated therewith the primary routing information field 904 for a particular routing code or bar code as described hereinabove. This information, once assembled, can then be transmitted to the subscriber 1312 and also be reflected in billing information and stored in the billing information database 1316.

Referring now to FIGURE 14, there is illustrated a flowchart depicting the operation for storing the profile for the user. The program is initiated in a block 1402 and then proceeds to a function block 1404, wherein the system will prompt for the profile upon initiation of the system. This initiation is a function that is set to activate whenever the user initially loads the software that he or she is provided. The purpose for this is to create, in addition to the setup information, a user profile. Once the user is prompted for this, then the program will flow to a decision block 1406 to determine whether the user provides basic or detailed information. This is selectable by the user. If selecting basic, the program will flow to a function block 1408 wherein the user will enter basic information such as name and serial number and possibly an address. However, to provide some incentive to the user to enter more information, the original prompt in function block 1404 would have offers for such things as coupons, discounts, etc, if the user will enter

additional information. If the user selects this option, the program from the decision block 1406 to a function block 1410. In the function block 1410, the user is prompted to enter specific information such as job, income level, general family history, demographic information and more. There can be any amount of information collected in this particular function block.

Once all of the information is collected, in either the basic mode or the more specific mode, the program will then flow to a function block 1412 where this information is stored locally. The program then flows to a decision block 1414 to then go on-line to the host or the ARS 308. In general, the user is prompted to determine whether he or she wants to send this information to the host at the present time or to send it later. If he or she selects the "later" option, the program will flow to a function block 1415 to prompt the user at a later time to send the information. In the disclosed embodiment, the user will not be able to utilize the software until the profile information is sent to the host. Therefore, the user may have to activate this at a later time in order to connect with the host.

If the user has selected the option to upload the profile information to the host, the program will flow to the function block 1416 to initiate the connect process and then to a decision block 1418 to determine if the connection has been made. If not, the program will flow along a "N" path to a time to decision block 1420 which will time to an error block 1422 or back to the input of the connect decision block 1418. The program, once connected, will then flow along a "Y" path from decision block 1418 to a function block 1428 to send the profile information with the ID of the computer or user to the host. The ID is basically, as described hereinabove, a "cookie" in the computer which is accessed by the program when transmitting to the host. The program will then flow to a function block 1430 to activate the program such that it, at later time, can operate without requiring all of the set up information. In general, all of the operation of this flowchart is performed with a "wizard" which steps the user through the setup process. Once complete, the program will flow to a Done block 1432.

Referring now to FIGURE 15, there is illustrated a flowchart depicting the operation of the host when receiving a transaction. The program is initiated at a start block 1502 and then proceeds to decision block 1504, wherein it is determined whether the system has received a

routing request, i.e., the routing information in the form of a bar code, etc., stored in the routing field 904 as described hereinabove with respect to FIGURE 9. The program will loop back around to the input of decision block 1504 until the routing request has been received. At this time, the program will flow along the "Y" path to a function block 1506 to receive the primary routing information and the user ID. Essentially, this primary routing information is extracted from the received field, in addition to the user ID. The program then flows to a function block 1508 to look up the manufacturer URL that corresponds to the received primary routing information and then return the necessary command information to the originating PC 108 in order to allow that PC to connect to the destination associated with the primary routing information. Thereafter, the program will flow to a function block 1510 to update the transaction database 1310 for the current transaction. In general, the routing information will be stored as a single field with the associated IDs. The profile database, as described hereinabove, has associated therewith detailed profiles of each user on the system that has activated their software in association with their ID. Since the ID was sent in association with the routing information, what is stored in the transaction database is the routing code, i.e., the barcode, in association with all of the IDs transmitted to the system in association with that particular routing code. Once this transaction database has been updated, as described hereinabove, the transactions can be transferred back to the subscriber at node 312 with the detailed profile information from the profile database 1308.

The profile information can be transmitted back to the subscriber or manufacturer in the node 312 in realtime or non-realtime. A decision block 1512 is provided for this, which determines if the delivery is realtime. If realtime, the program will flow along a "Y" path to a function block 1514 wherein the information will be immediately forwarded to the manufacturer or subscriber. The program will then flow to a function block 1516 wherein the billing for that particular manufacturer or subscriber will be updated in the billing database 1316. The program will then flow into an End block 1518. If it was non-realtime, the program moves along the "N" path to a function block 1520 wherein it is set for a later delivery and it is accrued in the transaction database. In any event, the transaction database will accrue all information associated with a particular routing code.

With a realtime transaction, it is possible for a manufacturer to place an ad in a magazine or to place a product on a shelf at a particular time. The manufacturer can thereafter monitor the times when either the ads are or the products are purchased. Of course, they must be scanned into a computer which will provide some delay. However, the manufacturer can gain a very current view of how a product is moving. For example, if a cola manufacturer were to provide a promotional advertisement on, for example, television, indicating that a new cola was going to be placed on the shelf and that the first 1000 purchasers, for example, scanning their code into the network would receive some benefit, such as a chance to win a trip to some famous resort in Florida or some other incentive, the manufacturer would have a very good idea as to how well the advertisement was received. Further, the advertiser would know where the receptive markets were. If this advertiser, for example, had placed the television advertisement in ten cities and received overwhelming response from one city, but very poor response from another city, he would then have some inclination to believe that either one poor response city was not a good market or that the advertising medium he had chosen was very poor. Since the advertiser can obtain a relatively instant response and also content with that response as to the demographics of the responder, very important information can be obtained in a relatively short time.

It should be noted that the disclosed embodiment is not limited to a single source PC 302, but may encompass a large number of source computers connected over a global communication network. Additionally, the embodiment is not limited to a single ARS 308 or a single advertiser server 312, but may include a large number of ARS and advertiser systems. It should also be noted that this embodiment is not limited only to global communication networks, but also may be used with LAN, WAN, and peer-to-peer configurations.

It should also be noted that the disclosed embodiment is not limited to a personal computer, but is also applicable to, for example, a Network Computer ("NetPC"), a scaled-down version of the PC, or any system which accommodates user interaction and interfaces to information resources.

Although the preferred embodiment has been described in detail, it should be understood that various changes, substitutions and alterations can be made therein without departing from the spirit and scope of the invention as defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method for associating a product with remote information for that product, comprising the steps of:

accessing a product identifier at a source location on a network which is in close association with a product such that it identifies the associated product;

appending routing information to the accessed product identifier, which routing information defines the location of a remote location on the network; and

transmitting the product information from the source location to the remote location over the network in accordance with the appended routing information, which routing information defines the network connection between the source location and the remote location on the network, wherein information regarding the associated product can be returned to the source location from the remote location.

2. The method of Claim 1, wherein the step of transmitting the product information is performed in response to the step of accessing.

3. The method of Claim 1, wherein the step of appending the routing information occurs in response to the step of accessing.

4. The method of Claim 1, wherein the routing information is associated with a secondary location on the network such that the step of transmitting to the remote location transmits the product information to the remote location in association with routing information for returning to the source location, and further comprising:

accessing a database of product routing information which provides an association between a predetermined product identifier and a remote product information location on the network, there being a plurality of such product routing information stored in the database;

accessing the database in response to receiving at the secondary location a transmitted product identifier from the source location;

This Sheet Has Been Replaced

comparing the received product identifier with the stored product routing information in the database;

if there is a match between the received product identifier and any of the stored product routing information, transmitting the matching product routing information back to the source location; and

at the source location, in response to receiving the matching product routing information, interconnecting the source location with the remote product information location over the network and receiving product information therefrom.

5. The method of Claim 4, and further comprising the step of displaying the received product information.

6. The method of Claim 1, wherein the product identifier is attached to the product.

7. The method of Claim 6, wherein the product identifier is a bar code.

8. The method of Claim 7, wherein the step of accessing comprises scanning the bar code.

9. The method of Claim 1, wherein the network is a global communication network that provides a universal resource locator (URL) for each location on the network and the routing information is comprised of the URL for the location.

10. The method of Claim 9, wherein a processing system is provided running a program that receives keystroke inputs for routing of information thereto and the step of appending routing information comprises the step of forming a keystroke string that emulates the combination of the routing information and the product identifier in keystrokes and transmits the emulated keystrokes to the program in the same manner that a user would key the strokes in, and the step of transmitting is performed in response to the appended routing information and product identifier transmitted to the program with an appended command associated therewith

that instructs the program to perform a transmit operation in a manner similar to a user interacting with the program to transmit routing information.

11. The method of Claim 10, wherein the appended routing information includes instructional information as to how the remote location is to handle the transmitted product identifier.

12. A method for interfacing a scanned product code with the product source, over a communication network, comprising the steps of:

inputting the scanned product code into an interface circuit at a source location;

appending routing information to the scanned product code to comprise a message packet;

in a first transaction, automatically routing the message packet through a computer operating a web browser at said source location, over said communication network to one or more remote locations; and

restructuring said message packet during each subsequent transaction of said first transaction, to define the direction of network communication between said source location and respective ones of one or more of the remote locations, and to link said scanned product code at said source location with said product source at one of said remote locations.

13. A method for interfacing a scanned product code with the product source, over a communication network, comprising the steps of: inputting said scanned product code into an interface circuit at a source location;

appending routing information to said scanned product code to comprise a first message packet; wherein said routing information defines network communication between said source location and a first remote location;

in response to said first routing information, automatically routing said first message packet from said source location over said communication network to said first remote location;

in response to said first message packet, automatically routing back to said source location, a second message packet comprising an address of a second remote location;

in response to said second message packet, automatically routing to said second remote location, a third message packet comprising a request for product information; and

in response to said third message packet, automatically routing to said source location, a fourth message packet comprising said product information.

14. A system for interfacing a scanned product code with the product source, over a communication network, comprising:

an interface circuit for receiving a scanned product code, said interface circuit appending routing information to said scanned product code to comprise a message packet;

a computer connected to said interface circuit and operating a web browser at a source location, said source location connected to one or more remote locations over a communication network;

wherein in a first transaction, said message packet is automatically routed through said computer by said web browser to one or more of said remote locations; and

wherein during each subsequent transaction, said message packet is restructured to define the direction of communication between said source location and respective ones of one or more of said remote locations, and to link said scanned product code at said source location with said product source at said remote location.

This Sheet Has Been Replaced